ICS 35.040
L 80

中华人民共和国国家标准

GB/T 18238.3—2002

目　　次

前　言

本标准等同采用国际标准ISO/IEC 10118-3:1998《信息技术 安全技术 散列函数 第3部分:专用散列函数》。

本标准附录A、附录B、附录C均为提示的附录。

本标准由中华人民共和国信息产业部提出。

本标准由中国电子技术标准化研究所归口。

本标准起草单位:中国电子技术标准化研究所。

本标准主要起草人:徐冬梅、张展新。

ISO/IEC 前言

ISO(标准化组织)和IEC(国际电工委员会)是世界性的标准化机构。国家成员体(都是ISO或IEC的成员国)通过国际组织建立的各个技术委员会参与制定针对特定技术领域的标准。ISO和IEC的各技术委员会在共同感兴趣的领域内进行合作。与ISO和IEC有联系的其他官方和非官方国际组织也可参与标准的制定工作。

对于信息技术领域,ISO和IEC建立了一个联合技术委员会,即ISO/IEC JTC1。由联合技术委员会提出的标准草案需分发给国家成员体进行表决。发布一项标准,至少需要75%的参与表决的国家成员体投票赞成。

国际标准ISO/IEC 10118-3是由ISO/IEC JTC1"信息技术"联合技术委员会的SC27"信息技术 安全技术"分委员会制定的。

ISO/IEC 10118在总标题"信息技术 安全技术 散列函数"下包含以下几个部分:

——第1部分:概述

——第2部分:采用n位块密码的散列函数

——第3部分:专用散列函数

——第4部分:采用模运算的散列函数

可能还会有后续部分。

本标准的附录A、附录B和附录C均为提示的附录。

中华人民共和国国家标准

信息技术　安全技术　散列函数
第3部分：专用散列函数

GB/T 18238.3—2002
idt ISO/IEC 10118-3:1998

Information technology—Security techniques—Hash-functions—Part 3:Dedicated hash-functions

1　范围

本标准规定了专用散列函数，即专门设计的散列函数。本标准的散列函数基于循环函数的迭代使用。本标准规定了三种不同的循环函数，从而产生了不同的专用散列函数。第一种和第三种提供了长度达160位的散列码，第二种提供了长度达128位的散列码。

2　引用标准

下列标准所包含的条文，通过在本标准中引用而构成为本标准的条文。本标准出版时，所示版本均为有效。所有标准都会被修订，使用本标准的各方应探讨使用下列标准最新版本的可能性。

GB/T 1988—1998　信息技术　信息交换用七位编码字符集(eqv ISO 646:1991)

GB/T 18238.1—2000　信息技术　安全技术　散列函数　第1部分：概述(idt ISO/IEC 10118-1:1994)

3　定义

GB/T 18238.1 中给出的定义以及下列定义适用于本标准：

3.1　**块**　block

长度为 L_1 的位串，即送往循环函数的第一个输入的长度。

3.2　**散列函数标识符**　hash-function identifier

标识特定散列函数的字节。

3.3　**循环函数**　round-function

把长度为 L_1 和 L_2 的两个二进制串变换成长度为 L_2 的一个二进制串的函数 $\phi(.\ ,.)$。它作为散列函数的一部分迭代使用，其中它把长度为 L_1 的数据串与前一步输出的长度为 L_2 的数据串组合起来。

3.4　**字**　word

一个32位的串。

4　符号和记法

本标准采用 GB/T 18238.1 中定义的符号和记法。

D	输入到散列函数的数据串。
H	散列码。
IV	初始化值。

中华人民共和国国家质量监督检验检疫总局 2002-07-18 批准　　2002-12-01 实施

L_X	位串 X 的长度(按位表示)。
$X \oplus Y$	位串 X 和 Y 的异或。

下列符号和记法适用于本标准:

a_i, a'_i	用于规定循环函数的索引序列。
B_i	字节。
C_i, C'_i	循环函数中使用的常数字。
D_i	填充处理后从数据串导出的块。
f_i, g_i	采用三个字作为输入并产生单个字作为输出,用于规定循环函数的函数。
H_i	在散列操作中用于存储中间结果的长 L_2 位的串。
L_1	送往循环函数 ϕ 的两个输入串中的第一个输入串的长度(按位表示)
L_2	送往循环函数 ϕ 的两个输入串中的第二个输入串的长度(按位表示),循环函数的输出串的长度(按位表示),以及 IV 的长度(按位表示)。
q	填充和分离过程后数据串中的块数。
$S^n()$	"循环左移" n 位操作,即,如果 A 是一个字并且 n 是一个非负整数,那么 S^n(A)表示将字 A 经过 n 次左循环移位而得到的字。
t_i, t'_i	用于规定循环函数的移位值。
W, X_i, X'_i, Y_i, Z_i	用来存储中间计算结果的字。
ϕ	一个循环函数,即,如果 X、Y 是长度分别为 L_1 和 L_2 的位串,那么 $\phi(X,Y)$ 是通过把 ϕ 应用于 X、Y 所得到的串。
$\wedge$	位串的逐位逻辑"与"操作,即,如果 A、B 是字,那么 $A \wedge B$ 是等于 A 和 B 逐位逻辑"与"得到的字。
$\vee$	位串的逐位逻辑"或"操作,即,如果 A、B 是字,那么 $A \vee B$ 是等于 A 和 B 逐位逻辑"或"得到的字。
$\neg$	位字符串的逐位逻辑非操作,即如果 A 是字,那么 $\neg A$ 是 A 逐位逻辑"非"得到的字
$\uplus$	模 2^{32} 加法操作,即,如果 A、B 是字,那么 $A \uplus B$ 是通过把 A 和 B 视为整数的二进制表示并计算它们的模 2^{32} 和而得到的字,其中结果被限制在 0 和 $2^{32}-1$ 之间,包括 0 和 $2^{32}-1$。
$:=$	表示在循环函数过程描述中所使用的"置等于"操作的符号,其中它表示符号左边的字应与符号右边表达的值相等。

5 要求

想要使用本标准中的散列函数的用户应选择:

- 以下规定的专用散列函数之一;以及
- 散列码 H 的长度 L_H。

注

1 定义了第一种和第二种专用散列函数以利于"小结尾"计算机软件的实现,也就是字中最低访问字节被解释为最低有效位;相反,第三种循环函数的定义便于"大结尾"计算机软件的实现,就是字的最低访问字节被解释为最高有效位。然而,通过适当地调整定义,任何循环函数都能够在"小结尾"计算机或者在"大结尾"计算机上实现。本文定义的全部散列函数把一个位串作为输入并且给出一个输出位串;这不依赖于每个散列函数内所使用的内部字节排序约定。

2 L_H 的选择影响散列函数的安全性。在进行 $2^{L_H/2}$ 散列码计算时被认为在计算上是不可行的环境中本标准所规定的全部散列函数被认为是无碰撞散列函数。

6 专用散列函数模型

6.1 概述

本标准中规定的散列函数要求使用循环函数 ϕ。后续各章规定了三种可替换函数 ϕ。

本标准中规定的散列函数提供长度为 L_H 的散列码，其中对于所使用的循环函数 ϕ 来说，L_H 的值小于或者等于 L_2。

在本标准散列函数的规范中，假设输入到散列函数的填充数据串是以字节序列形式表示的。如果所填充的数据串以 $8n$ 位序列形式，$x_0, x_1, \ldots, x_{8n-1}$ 表示，那么它将以以下的方式被解释为 n 字节的序列，$B_0, B_1, \ldots, B_{n-1}$。每组 8 个连续位被认为是一个字节，每组第一个位是该字节的最高有效位。因此，对任何 $i(0 \leqslant i < n)$：

$$B_i = 2^7 x_{8i} + 2^6 x_{8i+1} + \ldots + x_{8i+7}$$

对于本标准规定的三种专用散列函数中的每一种均定义了标识符。在第 7、8 和 9 章中规定的专用散列函数的散列函数标识符分别等于 31、32 和 33(十六进制)。34 到 3F(十六进制)之间的值保留作为以后的散列函数标识符使用。

6.2 散列操作

设 ϕ 为循环函数，IV 是长度为 L_2 的初始化值。对于本标准规定的散列函数，对给定的循环函数 ϕ，IV 的值应是固定的。

数据 D 的散列码 H 按照以下四步计算。

6.2.1 第 1 步(填充)

填充数据串 D 是以确保其长度是 L_1 的倍数。本标准后续各章中规定了填充法的特定实例。

6.2.2 第 2 步(分离)

数据串 D 的填充版被分离成 L_1 位的块 $D_1, D_2, \ldots\ldots, D_q$，其中 D_1 表示 D 填充后的第 1 个 L_1 位，D_2 表示第 2 个 L_1 位，依次类推。填充和分离过程如图 1 所示。

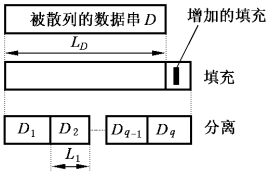

图 1 填充和分离过程

6.2.3 第 3 步(迭代)

设 $D_1, D_2, \ldots D_q$ 经填充和分离后，成为长度是 L_1 位的数据块。设 H_0 为与 IV 相等的位串。L_2 位串 $H_1, H_2, \ldots, H_q$ 用以下方法迭代计算：

对于 i 从 1 到 q：

$$H_i = \phi(D_i, H_{i-1})$$

迭代过程如图 2 所示。

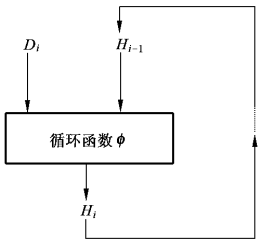

图 2 迭代过程

6.2.4 第 4 步(截短)

通过取最后长度为 L_2 位的输出串 H_q 的最左边的 L_H 位而导出散列码 H。

7 专用散列函数 1

注:本章包含循环函数的描述、初始化值以及附录 C 中[3]的 RIPEMD-160 填充法。

7.1 概述

本章规定了填充法、初始化值和循环函数,用于本标准描述的一般模型。这里规定的填充法、初始化值和循环函数,当用于上述一般模型时,要同时定义专用散列函数 1。该专用散列函数可用于包含至多 2^{64-1} 位的所有数据串 D。

用于专用散列函数 1 的 ISO/IEC 散列函数标识符等于 31(16 进制)。

7.2 参数、函数和常数

7.2.1 参数

该散列函数中,$L_1=512$,$L_2=160$。

7.2.2 字节排序约定

在第 7 章循环函数描述中,假定输入到循环函数中的块以字序列形式表示,每个 512 位的块由 16 个字组成。一个 64 个字节序列,$B_0,B_1,\ldots B_{63}$,应按照下列方法解释为 16 个字的序列,$Z_0,Z_1,\ldots,Z_{15}$。每组 4 个连续字节作为一个字,字的第 1 个字节是该字的最低有效字节。因此

$$Z_i = 2^{24}B_{4i+3} + 2^{16}B_{4i+2} + 2^{8}B_{4i+1} + B_{4i}(0 \leqslant i \leqslant 15)$$

把散列码从字序列转换到字节序列,应遵照相反的转换过程。

注:这里规定的字节排序与 9.2.2 中规定的字节排序不同。

7.2.3 函数

为便于软件实现,根据字操作来描述循环函数 ϕ。在本循环函数中使用一序列函数:$g_0,g_1,\ldots\ldots,g_{79}$,其中函数 $g_i(0\leqslant i\leqslant 79)$,以三个字 X_0、X_1 和 X_2 作为输入,并产生单个字作为输出。

函数 g_i 定义如下:

$g_i(X_0,X_1,X_2) = X_0 \oplus X_1 \oplus X_2 (0\leqslant i\leqslant 15)$;

$g_i(X_0,X_1,X_2) = (X_0 \wedge X_1) \vee (\neg X_0 \wedge X_2) (16\leqslant i\leqslant 31)$;

$g_i(X_0,X_1,X_2) = (X_0 \vee \neg X_1) \oplus X_2) (32\leqslant i\leqslant 47)$;

$g_i(X_0,X_1,X_2) = (X_0 \wedge X_2) \vee (X_1 \wedge \neg X_2) (48\leqslant i\leqslant 63)$;

$g_i(X_0,X_1,X_2) = X_0 \oplus (X_1 \vee \neg X_2) (64\leqslant i\leqslant 79)$

7.2.4 常数

本循环函数中使用两个常数字序列 $C_0,C_1,\ldots,C_{79}$ 和 $C'_0,C'_1,\ldots,C'_{79}$。用十六进制表示(其中最高有效位对应于最左边的位)定义如下:

C_i = 00000000 ($0\leqslant i\leqslant 15$);

C_i = 5A827999 ($16\leqslant i\leqslant 31$);

C_i = 6ED9EBA1 ($32\leqslant i\leqslant 47$);

C_i = 8F1BBCDC ($48\leqslant i\leqslant 63$);

C_i = A953FD4E ($64\leqslant i\leqslant 79$);

C'_i = 50A28BE6 ($0\leqslant i\leqslant 15$);

C'_i = 5C4DD124 ($16\leqslant i\leqslant 31$);

C'_i = 6D703EF3 ($32\leqslant i\leqslant 47$);

C'_i = 7A6D76E9 ($48\leqslant i\leqslant 63$);

C'_i = 00000000 ($64\leqslant i\leqslant 79$)。

本循环函数中使用了 80 个移位值组成的两个序列，其中每个移位值在 5 和 15 之间，用(t_0,t_1,...,t_{79})和(t'_0,t'_1,...,t'_{79})表示这两个序列。循环函数还使用 80 个索引组成的两个序列，其中每个序列中的每个值在 0 和 15 之间，用(a_0,a_1,...,a_{79})和(a'_0,a'_1,...,a'_{79})表示这两个序列。所有四个序列在下表中定义：

i	0	1	2	3	4	5	6	7
t_i	11	14	15	12	5	8	7	9
t'_i	8	9	9	11	13	15	15	5
a_i	0	1	2	3	4	5	6	7
a'_i	5	14	7	0	9	2	11	4

i	8	9	10	11	12	13	14	15
t_i	11	13	14	15	6	7	9	8
t'_i	7	7	8	11	14	14	12	6
a_i	8	9	10	11	12	13	14	15
a'_i	13	6	15	8	1	10	3	12

i	16	17	18	19	20	21	22	23
t_i	7	6	8	13	11	9	7	15
t'_i	9	13	15	7	12	8	9	11
a_i	7	4	13	1	10	6	15	3
a'_i	6	11	3	7	0	13	5	10

i	24	25	26	27	28	29	30	31
t_i	7	12	15	9	11	7	13	12
t'_i	7	7	12	7	6	15	13	11
a_i	12	0	9	5	2	14	11	8
a'_i	14	15	8	12	4	9	1	2

i	32	33	34	35	36	37	38	39
t_i	11	13	6	7	14	9	13	15
t'_i	9	7	15	11	8	6	6	14
a_i	3	10	14	4	9	15	8	1
a'_i	15	5	1	3	7	14	6	9

i	40	41	42	43	44	45	46	47
t_i	14	8	13	6	5	12	7	5
t'_i	12	13	5	14	13	13	7	5
a_i	2	7	0	6	13	11	5	12
a'_i	11	8	12	2	10	0	4	13

i	48	49	50	51	52	53	54	55
t_i	11	12	14	15	14	15	9	8
t'_i	15	5	8	11	14	14	6	14
a_i	1	9	11	10	0	8	12	4
a'_i	8	6	4	1	3	11	15	0

i	56	57	58	59	60	61	62	63
t_i	9	14	5	6	8	6	5	12
t'_i	6	9	12	9	12	5	15	8
a_i	13	3	7	15	14	5	6	2
a'_i	5	12	2	13	9	7	10	14

i	64	65	66	67	68	69	70	71
t_i	9	15	5	11	6	8	13	12
t'_i	8	5	12	9	12	5	14	6
a_i	4	0	5	9	7	12	2	10
a'_i	12	15	10	4	1	5	8	7

i	72	73	74	75	76	77	78	79
t_i	5	12	13	14	11	8	5	6
t'_i	8	13	6	5	15	13	11	11
a_i	14	1	3	8	11	6	15	13
a'_i	6	2	13	14	0	3	9	11

7.2.5 初始化值

对于该循环函数来说，初始化值 IV 应该总是下列 160 位的串，在这里表示为十六进制的 5 个字 Y_0、Y_1、Y_2、Y_3 和 Y_4 的序列，其中 Y_0 表示 160 位中最左边的 32 位。

Y_0= 67452301

Y_1= EFCDAB89

Y_2= 98BADCFE

Y_3= 10325476

Y_4= C3D2E1F0

7.3 填充法

需要填充数据串 D，使它具有 512 的整数倍位数。填充过程操作如下：

a) D 连接上 单个“1”。

b) 上一步所得的结果再连接上 0～511 个“0”位，以使结果串的(按位)长度与 448 模 512 同余。更确切地说，如果 D 的原始长度为 L_D，并设 r 为 L_D 除以 512 的余数，那么要连接的“0”的数目等于$447-r$(当 $r\leqslant 447$)或 $959-r$(当 $r>447$)。这样所得到的位串长度将比 512 位的整数倍少 64 位。

c) 把以 64 位二进制表示的 L_D 分成两个 32 位串，一个表示 L_D 的“最高有效串”，另一个表示“最低有效串”。把这两个 32 位的串(“最低有效串”在前)与前一步得到的字串相连接。

在以下对循环函数的描述中，每个 512 位数据块 $D_i(1\leqslant i\leqslant q)$ 视作为一个 16 个字的序列，Z_0，Z_1，…，Z_{15}，其中 Z_0 对应 D_i 最左边的 32 位。

7.4 循环函数的描述

循环函数 ϕ 操作如下。注意，在以下的描述中，使用符号 W、X_0、X_1、X_2、X_3、X_4、X'_0、X'_1、X'_2、X'_3、X'_4 来表示 11 个不同的字，这些字包含计算中要求的值。

a) 假定 ϕ 的 512 位(第一次)输入包含在 $Z_0, Z_1, \cdots, Z_{15}$ 中，其中 Z_0 是 512 位的最左边 32 位，同样假定 ϕ 的 160 位第二次输入包含在 5 个字 Y_0、Y_1、Y_2、Y_3 和 Y_4 中。

b) 置 $X_0 := Y_0 \quad X_1 := Y_1 \quad X_2 := Y_2 \quad X_3 := Y_3 \quad X_4 := Y_4$

c) 置 $X'_0 := Y_0 \quad X'_1 := Y_1 \quad X'_2 := Y_2 \quad X'_3 := Y_3 \quad X'_4 := Y_4$

d) 按照规定的次序执行以下四步，$i := 0$ 到 79：

(1) $W := S^{t_i}(X_0 \uplus g_i(X_1, X_2, X_3) \uplus Z_{ai} \uplus C_i) \uplus X_4$

(2) $X_0 := X_4 \quad X_4 := X_3 \quad X_3 := S^{10}(X_2) \quad X_2 := X_1 \quad X_1 := W$

(3) $W := S^{t'_i}(X'_0 \uplus g_{79-i}(X'_1, X'_2, X'_3) \uplus Z_{a'_i} \uplus C'_i) \uplus X'_4$

(4) $X'_0 := X'_4 \quad X'_4 := X'_3 \quad X'_3 := S^{10}(X'_2) \quad X'_2 := X'_1 \quad X'_1 := W$

e) 设

$W := Y_0$

$Y_0 := Y_1 \uplus X_2 \uplus X'_3$

$Y_1 := Y_2 \uplus X_3 \uplus X'_4$

$Y_2 := Y_3 \uplus X_4 \uplus X'_0$

$Y_3 := Y_4 \uplus X_0 \uplus X'_1$

$Y_4 := W \uplus X_1 \uplus X'_2$

f) 5 个字 Y_0、Y_1、Y_2、Y_3、Y_4 表示循环函数 ϕ 的输出。循环函数最终迭代后，通过使用 7.2.2 中规定的相反的转换过程，5 个字 Y_0、Y_1、Y_2、Y_3、Y_4 应转换成一个 20 个字节的序列，其中由 Y_0 产生前 4 个字节，Y_1 产生下 4 个字节，依次类推。这样，首(最左边的)字节对应于 Y_0 的最低有效字节，第 20 个(最右边的)字节对应于 Y_4 的最高有效字节。通过使用 6.1 中规定的相反的转换过程，这 20 个字节应转换成 160 个位的串，即首(最左边的)位对应于首(最左边的)字节的最高有效位，第 160(最右边的)位对应于第 20 个(最右边的)字节的最低有效位。

8 专用散列函数 2

本散列函数应仅用于当散列码包含小于或等于 128 位时就认为足够安全的应用中。

注：本章包含循环函数的描述、初始化值以及附录 C 中[3]的 RIPEMD-128 的填充法。

8.1 概述

本章规定了填充法、初始化值和循环函数，它们用于本标准描述的一般模型。这里规定的填充法、初始化值和循环函数，当它们用于上述一般模型时，要同时定义专用散列函数 2。该专用散列函数可适用于包含最多 $2^{64}-1$ 位的全部数据串 D。

用于专用散列函数 2 的 ISO/IEC 散列函数标识符等于 32(16 进制)。

8.2 参数、函数和常数

8.2.1 参数

本散列函数中，$L_1=512$，$L_2=128$。

8.2.2 字节排序约定

本散列函数的字节排序约定与第 7 章散列函数的约定相同。

8.2.3 函数

为便于软件实现，根据字操作来描述循环函数 ϕ。在本循环函数中使用一序列函数：$g_0, g_1, \ldots\ldots, g_{63}$，其中每个函数 $g_i(0 \leqslant i \leqslant 63)$，以三个字 X_0、X_1 和 X_2 作为输入，并产生单个字作为输出。

函数 g_i 与 7.2.3 中前 64 个函数的定义相同。

8.2.4 常数

本循环函数使用了两个常数字序列 $C_0,C_1,\ldots,C_{63}$ 和 $C'_0,C'_1,\cdots,C'_{63}$。用十六进制表示(其中最高有效位对应于最左边的位)它们定义如下:

C_i= 00000000 (0 ≤i≤ 15);

C_i= 5A827999 (16 ≤i≤ 31);

C_i= 6ED9EBA1 (32≤i≤ 47);

C_i= 8F1BBCDC (48≤i≤ 63);

C'_i= 50A28BE6 (0≤i≤ 15);

C'_i= 5C4DD124 (16≤i≤ 31);

C'_i= 6D703EF3 (32≤i≤ 47);

C'_i= 00000000 (48≤i≤ 63);

本循环函数也使用了两个 64 个移位值序列,其中每个移位值在 5 和 15 之间,用($t_0,t_1,\ldots,t_{63}$)和($t'_0,t'_1,\ldots,t'_{63}$)表示这些序列,它们的定义与 7.2.4 中相应序列的前 64 个值相同。

最后,本循环函数还使用两个 64 个索引序列,其中每个序列中的每个值在 0 和 15 之间,用($a_0,a_1,\ldots,a_{63}$)和($a'_0,a'_1,\ldots,a'_{63}$)表示这些序列,它们的定义与 7.2.4 中相应序列的前 64 个值相同。

8.2.5 初始化值

对于本循环函数来说,初始化值 IV 应总是下列 128 位的字符串,在这里表示为十六进制的 4 个字序列 Y_0、Y_1、Y_2 和 Y_3,其中 Y_0 表示 128 位中的最左边 32 位。

Y_0= 67452301

Y_1= EFCDAB89

Y_2= 98BADCFE

Y_3= 10325476

8.3 填充法

本散列函数使用的填充法与 7.3 中定义的填充法相同。

8.4 循环函数的描述

循环函数 ϕ 操作如下。注意,在以下的描述中,使用符号 W、X_0、X_1、X_2、X_3、X'_0、X'_1、X'_2、X'_3 来表示 9 个不同的字,这些字包含计算中要求的值。

a) 假定 ϕ 的 512 位第一次输入到中包含在 $Z_0,Z_1,\ldots,Z_{15}$ 中,其中 Z_0 是 512 位的最左边 32 位,同样假定第二次输入到 ϕ 中的 128 位包含在 4 个字 Y_0、Y_1、Y_2 和 Y_3 中。

b) 置 $X_0:=Y_0$ $X_1:=Y_1$ $X_2:=Y_2$ $X_3:=Y_3$

c) 置 $X'_0:=Y_0$ $X'_1:=Y_1$ $X'_2:=Y_2$ $X'_3:=Y_3$

d) 按照规定的次序执行以下四步,$i:=0$ 到 63:

(1) $W:=S^{t_i}(X_0 \uplus g_i(X_1,X_2,X_3) \uplus Z_{a_i} \uplus C_i)$

(2) $X_0:=X_3$ $X_3:=X_2$ $X_2:=X_1$ $X_1:=W$

(3) $W:=S^{t'_i}(X'_0 \uplus g_{63-i}(X'_1,X'_2,X'_3) \uplus Z_{a'_i} \uplus C'_i)$

(4) $X'_0:=X'_3$ $X'_3:=X'_2$ $X'_2:=X'_1$ $X'_1:=W$

e) 置

$W:=Y_0$

$Y_0:=Y_1 \uplus X_2 \uplus X'_3$

$Y_1:=Y_2 \uplus X_3 \uplus X'_0$

$Y_2:=Y_3 \uplus X_0 \uplus X'_1$

$Y_3:=W \uplus X_1 \uplus X'_2$

f) 4 个字 Y_0、Y_1、Y_2、Y_3 表示循环函数 ϕ 的输出。循环结束后，通过使用 7.2.2 中规定的相反的转换过程，4 个字 Y_0、Y_1、Y_2、Y_3 应转换成一个 16 个字节的序列，其中 Y_0 应产生得到前四个字节，Y_1 应产生下面 4 个字节，依次类推。这样首(最左边的)字节对应于 Y_0 的最低有效字节，第 16 个(最右边的)字节对应于 Y_4 的最高有效字节。通过使用 6.1 中规定的相反的转换过程，这 16 个字节应转换成 128 个位的串，即首(最左边的)位对应于首(最左边的)字节的最高有效位，第 128(最右边的)位对应于第 16 个(最右边的)字节的最低有效位。

9 专用散列函数 3

注：本章包含循环函数的描述、初始化值以及 SHA-1 的填充法。

9.1 概述

本条规定了填充法、初始化值和循环函数，用于本标准描述的一般模型。这里规定的填充法、初始化值和循环函数，当它们用于上述一般模型时，要同时定义专用散列函数 3。该专用散列函数可适用于包含最多 $2^{64}-1$ 位的全部数据串 D。

用于专用散列函数 3 的 ISO/IEC 散列函数标识符等于 33(16 进制)。

9.2 参数、函数和常数

9.2.1 参数

该散列函数中，$L_1=512$，$L_2=160$。

9.2.2 字节排序约定

在第 9 章循环函数的描述中，假定输入到循环函数的块以字序列形式表示，每个 512 位的块由 16 个这样的字组成。一个 64 个字节序列：$B_0, B_1, \ldots, B_{63}$，应按照下列方法，被解释为 16 个字序列：$Z_0, Z_1, \ldots, Z_{15}$。每组 4 个连续字节作为一个字，字的第 2 个字节是该字的最高有效字节。因此

$$Z_i = 2^{24} B_{4i} + 2^{16} B_{4i+1} + 2^8 B_{4i+2} + B_{4i+3} \quad (0 \leqslant i \leqslant 15)$$

把散列码从字序列转换到字节序列，应遵照相反的转换过程。

注：这里规定的字节次序与 7.2.2 中规定的字节次序不同。

9.2.3 函数

为便于软件实现，根据字操作来描述循环函数 ϕ。在循环函数中使用一序列函数：$f_0, f_1, \ldots, f_{63}$，其中每个函数 $f_i (0 \leqslant i \leqslant 79)$，以三个字 X_0、X_1 和 X_2 作为输入，产生单个字作为输出。

函数 f_i 定义如下：

$f_i(X_0, X_1, X_2) = (X_0 \wedge X_1) \vee (\neg X_0 \wedge X_2)$ $(0 \leqslant i \leqslant 19)$；

$f_i(X_0, X_1, X_2) = X_0 \oplus X_1 \oplus X_2$ $(20 \leqslant i \leqslant 39)$；

$f_i(X_0, X_1, X_2) = (X_0 \wedge X_1) \vee (X_0 \wedge X_2) \vee (X_1 \wedge X_2)$ $(40 \leqslant i \leqslant 59)$；

$f_i(X_0, X_1, X_2) = X_0 \oplus X_1 \oplus X_2$ $(60 \leqslant i \leqslant 79)$；

9.2.4 常数

该循环函数使用常数字序列 $C'_0, C'_1, \ldots, C'_{79}$。用十六进制表示(其中最高有效位对应于最左边的位)定义如下：

$C'_i =$ 5A827999 $(0 \leqslant i \leqslant 19)$；

$C'_i =$ 6ED9EBA1 $(20 \leqslant i \leqslant 39)$；

$C'_i =$ 8F1BBCDC $(40 \leqslant i \leqslant 59)$；

$C'_i =$ CA62C1D6 $(60 \leqslant i \leqslant 79)$。

9.2.5 初始化值

对于该循环函数来说，初始化值 IV 应总是 160 位的字符串，在这里表示为十六进制的 5 个字 Y_0、Y_1、Y_2、Y_3 和 Y_4，其中 Y_0 表示 160 位中最左边的 32 位。

$Y_0 =$ 67452301

Y_1= EFCDAB89

Y_2= 98BADCFE

Y_3= 10325476

Y_4= C3D2E1F0

9.3 填充法

需要填充数据串 D,使它具有 512 整数倍位数。填充过程操作如下:

a) D 连接上单个"1"。

b) 上一步所得的结果再连接上 0～511 个"0"位,以使结果串的(按位)长度与 448 模 512 同余。更确切地说,如果 D 的原始长度为 L_D,并设 r 为 L_D 除以 512 的余数,那么要连接的"0"的数目等于$447-r$(当 $r\leqslant 447$)或 $959-r$(当 $r>447$)。这样所得到的位串长度比 512 位的整数倍少 64 位。

c) 连接上一步结果串与 64 位二进制表示的 L_D,最高位在前。

在以下对循环函数的描述中,每个 512 位数据块 $D_i(1\leqslant i\leqslant q)$作为一个 16 个字序列:$Z_0$、$Z_1$、……$Z_{15}$,其中 Z_0 对应于 D_i 最左边的 32 位。

9.4 循环函数的描述

循环函数 ϕ 操作如下。注意,在以下的描述中,使用符号 W,X_0,X_1,X_2,X_3,X_4,Z_0,Z_1,…,Z_{79}来表示 86 个不同的字,这些字包含计算中要求的值。

a) 假定 ϕ 的 512 位(第一次)输入包含在 Z_0,Z_1,…,Z_{15}中,其中 Z_0 是 512 位的最左边 32 位,同样假定 ϕ 的 160 位(第二次)输入包含在 5 个字 Y_0、Y_1、Y_2、Y_3 和 Y_4 中。

b) 对于 $i=16$～79 置

$$Z_i := S^1(Z_{i-3}\oplus Z_{i-8}\oplus Z_{i-14}\oplus Z_{i-16})$$

c) 置 $X_0 := Y_0$　$X_1 := Y_1$　$X_2 := Y_2$　$X_3 := Y_3$　$X_4 := Y_4$

d) 执行以下两步,$i := 0$～79:

(1) $W := S^5(X_0) \uplus f_i(X_1,X_2,X_3) \uplus X_4 \uplus Z_i \uplus C'_i$

(2) $X_4 := X_3$　$X_3 := X_2$　$X_2 := S^{30}(X_1)$　$X_1 := X_0$　$X_0 := W$

e) 置

$Y_0 := Y_0 \uplus X_0$

$Y_1 := Y_1 \uplus X_1$

$Y_2 := Y_2 \uplus X_2$

$Y_3 := Y_3 \uplus X_3$

$Y_4 := Y_4 \uplus X_4$

f) 5 个字 Y_0、Y_1、Y_2、Y_3、Y_4 表示循环函数 ϕ 的输出。循环结束后,通过使用 9.2.2 中规定的相反的转换过程,5 个字 Y_0、Y_1、Y_2、Y_3、Y_4 应转换成一个 20 个字节的序列,其中 Y_0 应产生前 4 个字节,Y_1 应产生下面 4 个字节,依次类推。这样首(最左边的)字节对应于 Y_0 的最高有效字节,第 20 个(最右边的)字节对应于 Y_4 的最低有效字节。通过使用 6.1 中规定的相反的转换过程,这 20 个字节应转换成 160 个位的串,即首(最左边的)位对应于首(最左边的)字节的最高有效位,第 160(最右边的)位对应于第 20 个(最右边的)字节的最低有效位。

附 录 A
（提示的附录）
实 例

A1 概述

本附录给出了专用散列函数 1、2 和 3 的计算实例。对于每个散列函数都给出了散列码计算的九个例子。而且每个散列函数的例 3 和例 8 还给出了散列函数操作过程中得到的中间值。

A2 专用散列函数 1

本附录引用 GB 1988 编码的数据串。

注：附录 C 中的[3]含有专用散列函数 1 的伪码描述。

A2.1 例 1

本例中数据串为空串，即零长度串。

散列码是下列 160 位串：

9C 11 85 A5 C5 E9 FC 54 61 28 08 97 7E E8 F5 48 B2 25 8D 31

A2.2 例 2

本例中数据串由单个字节串组成，即字母“a”的 GB 1988 编码版本。散列码是下列 160 位串：

0B DC 9D 2D 25 6B 3E E9 DA AE 34 7B E6 F4 DC 83 5A 46 7F FE

A2.3 例 3

本例中数据串由“abc”GB 1988 编码版本的 3 字节串组成。它等同于位串：“01100001 01100010 01100011”。

经过填充过程后，由该数据串推导出的单个 16 字的块如下：

80636261 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000018 00000000

下列是变量 X_0、X_1、X_2、X_3、X_4、X'_0、X'_1、X'_2、X'_3、X'_4 的连续值（用 16 进制表示）：

67452301，EFCDAB89，98BADCFE，10325476，C3D2E1F0，67452301，EFCDAB89，98BADCFE，10325476，C3D2E1F0

C3D2E1F0，3115FC67，EFCDAB89，EB73FA62，10325476，C3D2E1F0，DDD63FB8，EFCDAB89，EB73FA62，10325476

10325476，B41192D5，3115FC67，36AE27BF，EB73FA62，10325476，322E7AE3，DDD63FB8，36AE27BF，EB73FA62

EB73FA62，3A35DC50，B41192D5，57F19CC4，36AE27BF，EB73FA62，883EE903，322E7AE3，58FEE377，36AE27BF

36AE27BF，D3786413，3A35DC50，464B56D0，57F19CC4，36AE27BF，92B2B79B，883EE903，B9EB8CC8，58FEE377

57F19CC4，0E946720，D3786413，D77140E8，464B56D0，58FEE377，F9091FF2，92B2B79B，FBA40E20，B9EB8CC8

464B56D0，D52BF632，0E946720，E1904F4D，D77140E8，B9EB8CC8，E5B09992，F9091FF2，CADE6E4A，FBA40E20

D77140E8，150BD8A8，D52BF632，519C803A，E1904F4D，FBA40E20，8B2D9FB3，E5B09992，247FCBE4，CADE6E4A

E1904F4D，3D6F601F，150BD8A8，AFD8CB54，519C803A，CADE6E4A，E755F422，8B2D9FB3，C2664B96，247FCBE4

519C803A，B7B60384，3D6F601F，2F62A054，AFD8CB54，247FCBE4，5922D09E，E755F422，B67ECE2C，C2664B96

AFD8CB54，B85A0A3F，B7B60384，BD807CF5，2F62A054，C2664B96，CF24E72C，5922D09E，57D08B9D，B67ECE2C

2F62A054，7F8B38E5，B85A0A3F，D80E12DE，BD807CF5，B67ECE2C，CA6A1C75，CF24E72C，8B427964，57D08B9D

BD807CF5，9DACA495，7F8B38E5，6828FEE1，D80E12DE，57D08B9D，227F6D84，CA6A1C75，939CB33C，8B427964

D80E12DE，BC05F46F，9DACA495，2CE395FE，6828FEE1，8B427964，5D801685，227F6D84，A871D729，939CB33C

6828FEE1，1494F053，BC05F46F，B2925676，2CE395FE，939CB33C，B3C3F4D5，5D801685，FDB61089，A871D729

2CE395FE，85861D02，1494F053，17D1BEF0，B2925676，A871D729，3D16242D，B3C3F4D5，005A1576，FDB61089

B2925676，597BF629，85861D02，53C14C52，17D1BEF0，FDB61089，FF459078，3D16242D，0FD356CF，005A1576

17D1BEF0，6347EF78，597BF629，18740A16，53C14C52，005A1576，927E40A8，FF459078，5890B4F4，0FD356CF

53C14C52，45C8FA44，6347EF78，EFD8A565，18740A16，0FD356CF，ACBB994E，927E40A8，1641E3FD，5890B4F4

18740A16，AD2956AF，45C8FA44，1FBDE18D，EFD8A565，5890B4F4，AD30AD24，ACBB994E，F902A249，1641E3FD

EFD8A565，5EAF16B7，AD2956AF，23E91117，1FBDE18D，1641E3FD，6261732E，AD30AD24，EE653AB2，F902A249

1FBDE18D，41730D4B，5EAF16B7，A55ABEB4，23E91117，F902A249，45ED27AF，6261732E，C2B492B4，EE653AB2

23E91117，FC0CCBD3，41730D4B，BC5ADD7A，A55ABEB4，EE653AB2，243C5668，45ED27AF，85CCB989，C2B492B4

A55ABEB4，042ECC93，FC0CCBD3，CC352D05，BC5ADD7A，C2B492B4，82F89BD1，243C5668，B49EBD17，85CCB989

BC5ADD7A，4D4D4377，042ECC93，332F4FF0，CC352D05，85CCB989，5FC74686，82F89BD1，F159A090，B49EBD17

CC352D05，5207002B，4D4D4377，BB324C10，332F4FF0，B49EBD17，B2720031，5FC74686，E26F460B，F159A090

332F4FF0，388278F5，5207002B，350DDD35，BB324C10，F159A090，58A100F8，B2720031，1D1A197F，E26F460B

BB324C10，62879D70，388278F5，1C00AD48，350DDD35，E26F460B，5992068B，58A100F8，C800C6C9，1D1A197F

350DDD35，A30A1FD9，62879D70，09E3D4E2，1C00AD48，1D1A197F，CC290DCA，5992068B，8403E162，C800C6C9

1C00AD48，BDA2B31B，A30A1FD9，1E75C18A，09E3D4E2，C800C6C9，863D625E，

CC290DCA，481A2D66，8403E162

09E3D4E2，F7211DEE，BDA2B31B，287F668C，1E75C18A，8403E162，6061B5A5，863D625E，A4372B30，481A2D66

1E75C18A，B6A665C6，F7211DEE，8ACC6EF6，287F668C，481A2D66，AA98ADB5，6061B5A5，F5897A18，A4372B30

287F668C，2D30FA02，B6A665C6，8477BBDC，8ACC6EF6，A4372B30，2999255A，AA98ADB5，86D69581，F5897A18

8ACC6EF6，C76D12F9，2D30FA02，99971ADA，8477BBDC，F5897A18，98237631，2999255A，62B6D6AA，86D69581

8477BBDC，516F84DF，C76D12F9，C3E808B4，99971ADA，86D69581，6C472A90，98237631，649568A6，62B6D6AA

99971ADA，F3FA5B05，516F84DF，B44BE71D，C3E808B4，62B6D6AA，2EAD5672，6C472A90，8DD8C660，649568A6

C3E808B4，D539625E，F3FA5B05，BE137D45，B44BE71D，649568A6，C5CB48BA，2EAD5672，1CAA41B1，8DD8C660

B44BE71D，D8500C99，D539625E，E96C17CF，BE137D45，8DD8C660，05286DFB，C5CB48BA，B559C8BA，1CAA41B1

BE137D45，7ECDE5B2，D8500C99，E5897B54，E96C17CF，1CAA41B1，88396DD2，05286DFB，2D22EB17，B559C8BA

E96C17CF，681D30B9，7ECDE5B2，40326761，E5897B54，B559C8BA，333F2212，88396DD2，A1B7EC14，2D22EB17

E5897B54，960F7BFD，681D30B9，3796C9FB，40326761，2D22EB17，C699295B，333F2212，E5B74A20，A1B7EC14

40326761，6770E498，960F7BFD，74C2E5A0，3796C9FB，A1B7EC14，BFD68874，C699295B，FC8848CC，E5B74A20

3796C9FB，75EB06C5，6770E498，3DEFF658，74C2E5A0，E5B74A20，BDDF3474，BFD68874，64A56F1A，FC8848CC

74C2E5A0，14FA827A，75EB06C5，C392619D，3DEFF658，FC8848CC，8CBC87E9，BDDF3474，5A21D2FF，64A56F1A

3DEFF658，804B0068，14FA827A，AC1B15D7，C392619D，64A56F1A，CDDA6EBF，8CBC87E9，7CD1D2F7，5A21D2FF

C392619D，475BA81B，804B0068，EA09E853，AC1B15D7，5A21D2FF，656C7DA3，CDDA6EBF，F21FA632，7CD1D2F7

AC1B15D7，D26BC25D，475BA81B，2C01A201，EA09E853，7CD1D2F7，76D66CA3，656C7DA3，69BAFF37，F21FA632

EA09E853，DBC5A2CB，D26BC25D，6EA06D1D，2C01A201，F21FA632，C9B17F72，76D66CA3，B1F68D95，69BAFF37

2C01A201，77367F5E，DBC5A2CB，AF097749，6EA06D1D，69BAFF37，65A60151，C9B17F72，59B28DDB，B1F68D95

6EA06D1D，8155A6B4，77367F5E，168B2F6F，AF097749，B1F68D95，33F3AC81，65A60151，C5FDCB26，59B28DDB

AF097749，C90C4D38，8155A6B4，D9FD79DC，168B2F6F，59B28DDB，9BFB827D，33F3AC81，98054596，C5FDCB26